

REMARKS/ARGUMENTS

Applicant thanks the Examiner for the interview of August 11, 2005, in which the Examiner acknowledged that the claims, as amended, overcame the cited prior art.

The Examiner rejects claims 68 and 88 under 35 U.S.C. §101 because the claimed subject matter is directed to non-statutory subject matter. Claims 68 and 88 have been amended to overcome this rejection.

The Examiner rejects claims 40, 43-46, 48, 69 and 72 under 35 U.S.C. §103(a) as being unpatentable over Jordan et al. (hereinafter Jordan) (U.S. 6,438,652) in view of Applicant's Admitted Prior Art (AAPA); claims 47 and 73 under 35 U.S.C. §103(a) as being unpatentable over Jordan in view of AAPA and further in view of Balijepalli et al. (hereinafter Balijepalli) (U.S. 2004/0230566); claims 41-42, 49, 50-53, 55-56, 70-71, 74, and 76-79 under 35 U.S.C. §103(a) as being unpatentable over Jordan in view of AAPA and further in view of Wallace, Jr (U.S. 2002/0112154); claim 81 under 35 U.S.C. §103(a) as being unpatentable over Jordan in view of AAPA and further in view of Wallace, Jr, and further in view of Balijepalli; claim 61 under 35 U.S.C. §103(a) as being unpatentable over Jordan in view of Balijepalli; claims 63-67 and 83-87 under 35 U.S.C. §103(a) as being unpatentable over Jordan in view of Wallace, Jr; and claims 54, 57-60, 62, 75, 80, 82, and 88 under 35 U.S.C. §102(e) as being anticipated by Jordan.

Applicant respectfully traverses the Examiner's rejections. The cited references fail to teach or suggest at least the following italicized features of the pending independent claims:

40. An arrangement for serving information requests, comprising:
a plurality of information servers connected to a communications network, all of the information servers having a common address on the communications network and serving a set of information to clients, each of the information servers being configured to receive a transaction request associated with an individual transaction and to provide a response to each transaction request; and
a content director connecting the information servers to the communications network and distributing transaction requests among the information servers comprising:
a flow switch that selects an appropriate information server to service each transaction request and thereafter forwards at least portions of the transaction request to a selected one of the information servers;
a cache processor that accesses a plurality of objects in response to communications received from the flow switch;
a cache that stores, in a hot invariant table, the plurality of objects corresponding to at least some of the transaction requests forwarded to one or more of the information servers, the hot invariant table identifying information frequently requested from the information servers and including, for each invariant identifying corresponding information, a hit counter indicating a number of transaction requests, received by the plurality of servers over a determined time interval, requesting the corresponding information;
a digest generator that generates, when the hit counter for an invariant indicates at least a threshold transaction request receipt frequency but not when the hit counter fails to indicate at least a threshold transaction receipt frequency, a digest value pointing to the location in the hot invariant table where the corresponding entry is stored; and
a digest store that stores the digests corresponding to frequently requested content.

54. In an arrangement comprising a plurality of information servers connected to a communications network, each of the information servers being configured to receive a transaction request associated with an individual transaction and to provide a response to each transaction request, a method for serving transaction requests from clients comprising:
maintaining a hot invariant table identifying information frequently requested from the information servers, the hot invariant table including, for each invariant identifying corresponding information, a hit counter indicating a number of transaction requests, received by the plurality of information servers over a determined time interval, requesting the corresponding information;
generating, when the hit counter for a selected invariant indicates at least a threshold transaction request receipt frequency, a digest value pointing to the location in

*the hot invariant table where the entry corresponding to the selected invariant is stored;
and*

*accessing a digest store comprising the digest values to select an information
server to service a transaction request for frequently requested information.*

69. An arrangement for serving information requests, comprising:

a plurality of information servers connected to a communications network, all of the information servers having a common address on the communications network and serving a set of information to clients, each of the information servers being configured to receive a transaction request associated with an individual transaction and to provide a response to each transaction request; and

content director means for connecting the information servers to the communications network and distributing transaction requests among the information servers comprising:

first flow switching means for parsing plain text transaction requests to locate selected fields, selecting an appropriate information server to service each transaction request, and thereafter forwarding at least portions of the parsed transaction requests to a selected one of the information servers;

cache processing means for accessing a plurality of objects in response to communications received from the first flow switching means;

cache means for storing, in a hot invariant table, the plurality of objects corresponding to transaction requests forwarded to one or more of the information servers, the hot invariant table identifying information frequently requested from the information servers and including, for each invariant identifying corresponding information, a hit counter indicating a number of transaction requests, received by the plurality of information servers over a determined time interval, requesting the corresponding information;

digest generator means for generating, when the hit counter for an invariant indicates at least a threshold transaction request receipt frequency but not when the hit counter fails to indicate at least a threshold transaction receipt frequency, a digest value pointing to the location in the hot invariant table where the corresponding entry is stored; and

digest store means for storing the digests corresponding to frequently requested content.

75. In an arrangement comprising a plurality of information servers connected to a communications network, each of the information servers being configured to receive a transaction request associated with an individual transaction and to provide a response to each transaction request, a method for serving transaction requests from clients comprising:

maintaining a hot invariant table identifying information frequently requested from the information servers, the hot invariant table including, for each invariant identifying corresponding information, a hit counter indicating a number of transaction requests, received by the plurality of servers over a determined time interval, requesting the corresponding information;

when the hit counter for an invariant indicates at least a threshold transaction request receipt frequency, locating the information associated with the invariant at a cache information server and thereafter directing a transaction request for information associated with the invariant to a cache information server; and

when the hit counter for an invariant falls below a threshold transaction request receipt frequency, directing a transaction request for information associated with the invariant to an origin information server.

Jordan

Jordan is directed to a server farm including cooperating cache servers in which information requests are forwarded to a cooperating cache server if the requested object cannot be found locally. As a consequence of direct and forwarded information requests, cache servers can become overloaded. Load balancing is effected by a load monitor 120 that shifts one or more of the forwarded requests for the object between cooperating cache servers based on a load condition and a forwarding frequency for the object. The load condition is a weighted sum of a count of the forwarded requests and a count of direct requests to the cooperating cache server. Overloaded servers are those with loads exceeding a threshold.

Jordan is distinguishable from the present invention for a number of reasons. Because Jordan is directed to monitoring and responding to the load condition of each cooperating cache server *individually*, the load condition is determined on a server-by-server basis and *not* for the servers as a collective group. Moreover, Jordan monitors the request frequency not for specific invariants which are associated with specific information but rather for specific cache servers,

each of which contains information associated with a number of different invariants. There is thus no motivation or suggestion to modify Jordan to monitor the load condition for the group of cooperating cache servers because Jordan is specifically directed to load balancing caused by requests forwarded by one cache server to another cache server. Monitoring load conditions for the group of servers would fail to recognize the effects of forwarded requests.

Cooperating cache servers can periodically exchange and maintain one or more of the load condition information, the forwarding frequency, and the ownership of information. An overloaded cooperating cache server can identify a less loaded cooperating cache server and communicate a shift request and a copy of the cached object to the less loaded cooperating cache server (which then caches the object) so that subsequent requests for the object will not be forwarded. As noted at col. 5, lines 61-63; col. 5, line 66 to col. 6, line 2; col. 6, lines 54-64; and col. 7, lines 7-15, the load monitor 120 does not consider the "hotness" of the requested information when no load imbalance condition or loading trend is found to exist. The load table 102 includes the load condition 1021 of each cache server 150 so that overloaded and underloaded servers can be identified. The caching table 1010 includes the forwarding frequency 1011 and ownership 1012 information of an object or a partition of objects.

Jordan also discloses the use of a hash function to locate information. In the Cache Array Routing Protocol, the entire object space is partitioned among the cooperating cache servers, with one partition for each server. When a request is received by a cache server, the hash function is applied to a key from the request, such as the URL or the destination IP address, to

identify the partition. If the hash partition is that assigned to the requesting server, the request is serviced locally. Otherwise, it is forwarded to the proper cache server in the identified partition. The hash function is used on all object requests and not simply for object requests when the hit counter for an invariant associated with the requested object indicates at least a threshold transaction receipt frequency.

Jordan fails to teach or suggest the italicized features above. Depending on the independent claim, these features include: (a) the maintenance of a “hot table” based not on the number and/or frequency of requests for objects from a selected server but on the number and/or frequency of requests *received by a group of servers* for the objects; (b) the generation, *when a hit counter for an invariant indicates at least a threshold transaction request receipt frequency*, of a digest value pointing to the location in the hot invariant table where the corresponding entry is stored; and/or (c) a digest store that stores the digests corresponding to frequently requested content.

For the reasons stated below, these deficiencies are not overcome by the other cited references.

Wallace, Jr

Wallace, Jr., is directed to establishing secure network user states. The states are secured by creating a first key from a received user key, encrypting user data with the key, storing the encrypted user data in a cookie, and sending the cookie to the computer, such that subsequently a secure user state between the server and user is established by receiving the cookie and the user

key from the computer. The computer in response creates a second key matching the first key, decrypts, using the second key, encrypted user data extracted from the cookie, and establishes the secure state based on the decrypted user data. The cookie contains user-specific information, such as the user's identity, when the user has visited a selected website, user's IP address, mailing address, email address, age, sex, credit card information, login/password combinations, preferences, hobbies, education level, browsing (click) history, browsing history with click frequency, browsing preferences, assigned primary keys, assigned GUIDs, etc. (Page 3, ¶0054.)

Balijepalli

Balijepalli et al. is directed to a web-based multi-user system and method for identifying, retrieving, and delivering information corresponding to items contained in a user search list from two or more information sources on the World Wide Web. The system includes a central server that periodically searches the information sources on the web for information corresponding to the items contained in the user search list. The central server retrieves the information onto a storage database where a determination is made as to whether the information is current. The central server electronically delivers only the current information to the user. The system and method therefore provides to the user automatic and periodic electronic reports containing only current or updated information corresponding to items contained in the user search list that the user desires to track.

Accordingly, the pending claims are allowable.

The dependent claims provide further reasons for allowance.

By way of example, dependent claim 41 requires the flow switch to parse plain text transaction requests to locate selected fields and a cryptographic module in the flow switch to decrypt, prior to parsing and information server selection by the flow switch, cipher text transaction requests and provide plain text transaction requests to the flow switch, wherein, prior to decryption, the cipher text transaction requests have not been routed by another flow switch. (See also claims 55, 70, and 76). Although Wallace, Jr does teach parsing an encrypted payload, Wallace, Jr fails to teach or suggest parsing *plain text* transaction requests prior to routing of the request by the flow switch.

Dependent claim 42 is directed to the receipt of first and second encrypted transaction requests from different clients having a common electronic address and served substantially simultaneously by different information servers, wherein at least some of the responses include a cookie, wherein the cookie is generated by the information server previously assigned by the flow switch to service transaction requests from the client, and wherein the flow switch uses at least one of an invariant, a cookie, and a tag in the parsed plain text equivalent of each transaction request to select an appropriate information server to service each of the first and second transaction requests. (See also claims 56, 71, and 77.) Dependent claim 43 requires each invariant in the hot invariant table to have a corresponding timestamp indicating when the respective entry was last updated, and a tag identifying a corresponding information server providing the corresponding information. (See also claim 57 and 80). Dependent claim 50 requires the flow switch to tag a transaction response, the tag identifying an information server

generating the transaction response. (See also claims 64 and 84.) Dependent claim 51 requires at least some of the responses to include a cookie, wherein the cookie is generated by the information server previously assigned by the flow switch to service transaction requests from the client, and wherein the cookie is different from the tag. (See also claims 65, 74, and 85.) Dependent claim 52 requires the tag to be concatenated to the cookie. (See also claims 66 and 86.) Wallace fails to teach or suggest the use of a server identifying tag. Wallace does teach the use of a cookie but fails to teach the use of a tag in addition to the cookie. As defined in the *Dictionary of Computer and Internet Terms*, a “cookie” is “information stored on a user’s computer by a WEB BROWSER at the request of software at a web site. *Web sites use cookies to recognize users who have previously visited them.*” Thus, cookie’s do not recognize information servers but visitors to the information servers. For at least this reason, the mere teaching of a cookie fails to render obvious the use of a server-specific tag to identify a server and provide fast switching of request packets to the appropriate server.

Dependent claim 53 states that, during a first time interval, the flow switch is in a tagging mode in which the switch generates and appends tags to transaction responses and, during a second different time interval, the switch operates in a digesting mode in which digests are generated, invariant hotness is monitored, and transaction requests are routed to information servers based on requested invariant hotness and/or cookie. At col. 5, lines 61-63; col. 5, line 66 to col. 6, line 2; col. 6, lines 54-64; and col. 7, lines 7-15, Jordan teaches only that the load monitor 120 does not consider the “hotness” of the requested information when no load

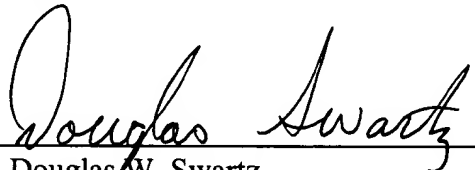
*Application No. 09/921,458
Reply to Office Action of June 21, 2005
Amendment dated August 20, 2005*

imbalance condition or loading trend is found to exist. Jordan fails to teach the use of separate tagging and digesting modes. (See also claims 67 and 87.) Jordan fails to teach the use of transaction response tagging, let alone the performance of tagging or hotness monitoring in one mode but not in another mode.

Based upon the foregoing, Applicants believe that all pending claims are in condition for allowance and such disposition is respectfully requested. In the event that a telephone conversation would further prosecution and/or expedite allowance, the Examiner is invited to contact the undersigned.

Respectfully submitted,

SHERIDAN ROSS P.C.

By: 
Douglas W. Swartz
Registration No. 37,739

1560 Broadway, Suite 1200
Denver, Colorado 80202-5141
(303) 863-9700

Date: August 18, 2005